University of Virginia

Charles L. Brown Department of Electrical and Computer Engineering
THE INNOVATION TEAM - Project proposal
Report to PICo Review Board by
Mark Cheung, Austin Moran, Xiafei Yang

SUBJECT: This report proposes a solution methodology for designing a 1Mb low-power SRAM to meet PICo's specifications. Our primary goal is to minimize the total power with reasonable sizing and delay. Our secondary goal is to consider as many special feature as possible by looking at the array of research in this area.

Introduction

A typical structure for an SRAM contains decoders, memory array, sense amplifier and periphery circuit to access the bit cells. Energy will be consumed by all these components. The key metric to optimize energy consumption is (Active Energy per Access)^2*Delay*Area*Idle Power. In order to win the contract of Portable Instruments Company (PICo), we have designed an SRAM focused on low-power techniques.

Approach

There are three different components of energy dissipation in an SRAM [5]

1. the dynamic energy to switch the capacitance in the decoders, bitlines, datalines and other control signals within the array;
2. the energy of the sense amplifiers;
3. the energy loss due to the leakage currents.

In order to significantly reduce the power consumption in SRAM, the following techniques are developed.[7]

- Capacitance reduction of wordlines, bitlines and decoders
- Current reduction for wordlines, periphery circuits and sense amplifier
- Operating voltage reduction
- Leakage current reduction by utilizing multiple threshold voltage technology(MT-CMOS)

The large capacitive elements in a memory are wordline, bitlines and datalines each with a number of cells connected to them. So reducing the size of these lines can have a great impact on capacitance reduction and therefore save the dynamic energy to switch capacitance.[7] For this purpose, the decoder will employ the divided word line structure, where part of the address is decoded to access the horizontal global word line and the remaining address bits activate the vertical block select line. The intersection of these two activates the local word line. The cells connected to this word line transfer their data onto the bit lines. Data from a subset of bit lines is routed by the column mux into the sense amplifiers which amplify and drive it onto the data lines.[5] Figure 1 is the SRAM model that shows the above arrangements.

See figure 1 in appendix

Partition

For large SRAMs, significant improvements in delay and power can be achieved by partitioning the cell array into smaller sub arrays, rather than having a single monolithic array. Typically, a large array is partitioned into a number of identically sized sub arrays (blocks), each of which stores a part of the accessed word, and all of which are activated simultaneously to access the complete word.[6]

To justify the reason why we split the entire memory array into blocks of sub-arrays, let us take a 64 X 64 array for an example. Consider C*j* as a parasitic capacitance associated with a single bitcell load on a bitline and C*ch* as a parasitic capacitance associated with a single bitcell on the wordline, then the total bitline capacitance is 64*C*j* and the total word capacitance is 64*C*ch.* If this array is divided into four isolated blocks of 32 X 32 bitcells, the total bitline and wordline capacitances of each block would be halved. Thus, the total capacitance per read/write that would need to be discharged or charged is given 1024*C*j* + 32*C*ch* for the partition structure as opposed to 4096*C*j* + 64*C*ch* for the 64 X 64 array. However, this technique carries a penalty due to additional decode and control logic and routing. [9][10]

So in order to get a balance of capacitance reduction by partition and the additional decode and control logic and routing incurred, we will get further tests to collect the data of different numbers of rows and columns per block to optimize the energy and delay.

Figure 2 shows five cases for energy and delay tested by a previous group. According to metric,

$$Metric = (Active\ Energy\ per\ Access)^2 * Delay * Area * Idle\ Power$$

we will choose dimensions of 256 rows, 64 columns, and 64 blocks for the present design. But further tests will be done when taking decoding components and other peripheral circuit into consideration.

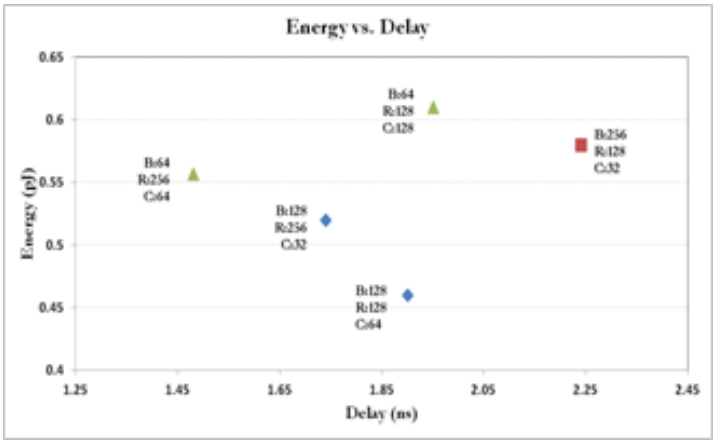


Figure 6. Energy v. Delay of the optimal points

Sense Amplifier

Sense amplifiers are used to detect the voltage difference between a bit line pair that develops during the read operation of SRAM. There exist two different methods of sensing this voltage difference: voltage sense amplifiers which read the voltages of a bit line pair and current sense amplifiers which detect the current of a discharging bit line.

DH Huang derives an expression for the sense delay (time it takes to detect a change in the bit line pair) based on the distributed resistance-capacitance model of a bit line. The equation is as follows [8] :

$$Delay = (R_T C_T)/2 * (R_B + R_T/3 + R_L)/(R_B + R_T + R_L) + R_B C_T * (R_L)/(R_B + R_T + R_L)$$

R_T= total resistance of the bitline,
R_B= output resistance of the bitline load in parallel with the drain resistance of access transistor,
R_L= load resistance of the sense amplifier,
C_T= total capacitance of the bitline

For a voltage sense amplifier the resistance of the sense amplifier (R_L) is assumed to be infinite (ideally). This simplifies the equation for delay to [8]:

$$Delay = (R_T C_T)/2 * (1 + 2R_B/R_T)$$

For a current sense amplifier the resistance of the sense amplifier (R_L) is assumed to be zero (ideally). This simplifies the equation for delay to [8]:

$$Delay = (R_T C_T)/2 * (R_B + R_T/3)/(R_B + R_T)$$

[8] provides example values for the variables to show that current amplifiers have much less delay. R_T = 250ohm, R_B= 2.5kohm, C_T= 2pF. Substituting these values into the equations for delay gives:

Delay of voltage sense amp = 5.25ns
Delay of current sense amp = 0.235ns

These results show that the delay from current source amplifiers is less than the delay from a voltage source amplifier. Having a smaller delay on the sense amplifier is important for several reasons. Firstly, delay is a part of the final design metric. Minimizing delay then should help us achieve a higher score. Second, a smaller delay from the sense amplifier means that the bitline has less time to discharge. This means that less energy is wasted discharging the bit line to a level where it can be detected with a sense amplifier.

So, a current sense amplifier is more efficient in both delay and energy usage compared to a voltage sense amplifier. However there exist several variations of a current sense amplifier outlined in [8].

We will be using the hybrid current sense amplifier as it combines the 4T current SA with the clamped bit line sense amplifier. The effect of bit line clamping is to make the input to the sense amplifier low impedance which limits the voltage swing on the bit lines [8]. As well, the output of the sense amplifier does not see the capacitance of the bit lines and thus is able to change quicker, reducing delay and energy usage. The hybrid current sense amplifier also has a current conveyor as part of its circuitry. This equalizes the voltage between the two floating lines feeding the sense amp. This voltage difference comes from repeatedly reading the same bit line values and can cause errors in

the output of the sense amp [8].

Decoders
There are three decoders, for each row of blocks, each word per row and each block. To reduce the capacitance of the address lines to a row decoder and the word-line RC delay, a two-stage hierarchical row decoder is adopted. Row decoder is controlled by the 8-bit address to drive 256 word lines. An AND gate is used with the word line outputs of the decoder and a block select bit from a 6:64 decoder to determine which of the 64 blocks to access at a time, which represents the predecoding and final decoding stages.[7]

LAYOUT

For the layout (figure 2 in appendix) of the 6T SRAM cell we have decided on layout #5 proposed in [2]. This layout (#5) differs from the layout most commonly used (#4) in that the gates are now in line with the inputs to the cell [2]. This does result in an increase of the cell area but also has the effect of decreasing the capacitance of the bitline. In [2] the authors state that "An improvement of 13% or more in read access delay may be realized due to reduction in bit line length." but also that (in comparison to #4) "Based on this analysis, an area penalty of approximately (18-40%) will need to be weighed against the advantages of…(#5)." For our design metric the increase in area should be more than made up for by the savings in delay and energy usage (due to energy being squared). In fact, according to the metric, a delay decrease of 13% and an area increase of ~20% would only need an energy savings of 6% to break even. If the new layout directly or indirectly (due to SA changes) results in an energy usage decrease of 6% (will be tested for), it will be used in our design.

The savings in energy will come from the reduced bit line capacitance. It will reduce the energy required to charge the bit line (a function of both voltage and capacitance) and allow the sense amp to activate earlier (lower capacitance results in a lower time constant = BL,BLB change faster).

SPECIAL FEATURES

Minimum VDD: [3]Because the key metric for our design is (Total Power)^2*Delay*Area*Idle Power, we plan on reducing V_{DD} as much as possible because it plays a large role in reducing power, which is also squared in the key metric. This will deal with reducing VDD from the standard 2.5V to some threshold that keeps V_{DD} low, but still allows the SRAM to operate correctly.

Multi-Vt: A significant cause of leakage power is due to subthreshold leakage current, which has a direct dependence on Vt: Ileakage= Io (W/L)10^(Vgs-Vt)/S. As Vt decreases, subthreshold leakage increases exponentially. [1] suggests using multiple-threshold CMOS circuits, specifically transistor with higher threshold voltage (NMOS_VTH and PMOS_VTH models). The trade-offs include reduction in gate's drive strengths that results in increased delay. Consequently, we intend to apply this method only on non-critical paths so as to not increase the critical path delay.
See appendix (table 1, figure 3)

Sleep Mode (also known as Gated-Vdd in [1]): [4][7]This feature is to help reduce the idle power and reduce leakage current by implementing multi-threshold voltage CMOS circuit, and therefore optimize the metric. The parts of the memory that have this component will be turn off whenever

sleep mode is activated.

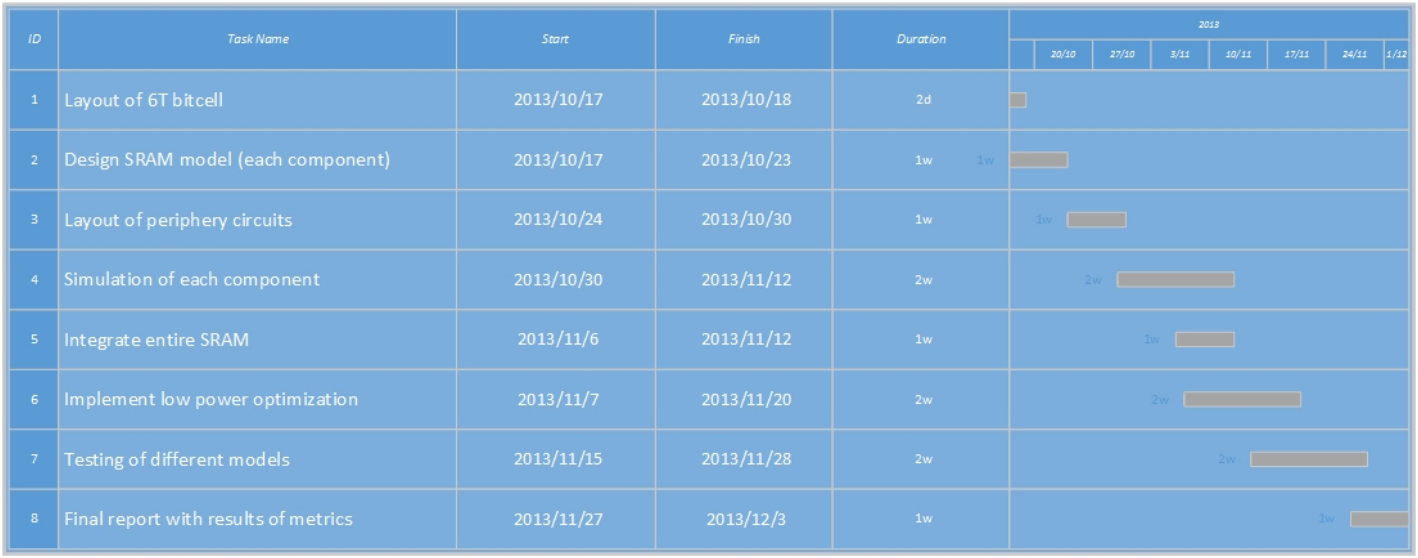
Data Retention Gated-ground (DRG) [1]: Similar to sleep mode, gated-grounded technique uses a transistor to connect the actual ground to whatever was connected to it before. We can then simply connect the word line to the gate of this transistor and reduce leakage current. The disadvantages are that it is less tolerable to noise with smaller SNM margin and creates more delay. Careful sizing is also required.

Error Correcting Code: [3], [4]Error Correcting Code is a global feature for error detection which will increase the accuracy and efficiency of data processing. This feature here does not necessarily lead to a better metric result but a better user-friendly perspective.

Ideas we also considered: Gate-source self-reverse biasing: [3], 4T bit cell, low-power MUX, parallel access, wire width modulation, drowsy SRAMs

TIMELINES

ID	Task Name	Start	Finish	Duration
1	Layout of 6T bitcell	2013/10/17	2013/10/18	2d
2	Design SRAM model (each component)	2013/10/17	2013/10/23	1w
3	Layout of periphery circuits	2013/10/24	2013/10/30	1w
4	Simulation of each component	2013/10/30	2013/11/12	2w
5	Integrate entire SRAM	2013/11/6	2013/11/12	1w
6	Implement low power optimization	2013/11/7	2013/11/20	2w
7	Testing of different models	2013/11/15	2013/11/28	2w
8	Final report with results of metrics	2013/11/27	2013/12/3	1w



TASK BREAKDOWN

Austin Moran

- Design and Simulation (Sense Amplifier, write amplifier)
- Word line optimization
- Clock &Timing
- Schematics

Mark Cheung

- Design and Simulation (6T Bitcell, inverter, MUX)
- Process Corner Simulation
- Layout Design
- Overall Simulation Results

Xiafei Yang

- Design and Simulation(Address Decoders)
- Block Optimization
- Wiki Update
- Project Journal

REFERENCE

[1] Jacob, B., Ng, S., & Wang, D. (2008). *Memory systems : cache, DRAM, disk.* San Francisco, California: Morgan Kaufmann .

[2] Mann, R., Calhoun, B. (2011). *New category of ultra-thin notchless 6T SRAM cell layout topologies for sub-22nm .* 12th International Symposium on Quality Electronic Design

[3] Itoh, K., Horiguchi, M., Tanaka, H. (2007) *Ultra-low voltage nano-scale memories.* New york, New york: Springer Science+Business Media

[4] Bailey, S., Linger, K., Lorenzo, R., & Thompson, J. (2011). Team 1 implementation of a low power SRAM design using 45 nm FreePDK technology. Unpublished.

[5] Bharadwaj S. Amrutur and Mark A. Horowitz.(2000) *Speed and Power Scaling of SRAM's.* IEEE TRANSACTIONS ON SOLID-STATE CIRCUITS, VOL. 35, NO. 2, FEBRUARY 2000

[6] Bharadwaj S. Amrutur (1999) *Design and Analysis of Fast Low Power SRAMs.* A Dissertation submitted to the department of Electrical Engineering and the Committee on Graduate Studies of Stanford University in partial fulfillment of the requirements for the degree of Doctor of Philosophy

[7] Martin Margala. (1999) *Low-Power SRAM Circuit Design.* Records of the IEEE International Workshop on Memory Technology, Design and Testing, August 9-10, 1999 San Jose, California

[8] Der-Chen Huang. (No Date Listed) *Sense Amplifier for SRAM.* http://soc.cs.nchu.edu.tw/upload_data/Sense%20Amplifier%20for%20SRAM.pdf

[9] J. S. Caravella, *A 0.9V, 4K SRAM for Embedded Applications.* in Proceedings of CICC, pp.119-122, May 1996

[10] J. S. Caravella, *A Low Voltage SRAM For Embedded Applications*. IEEE Journal of Solid-State Circuits, vol. 32, no. 3, pp. 428-432, March 1997

Appendix:

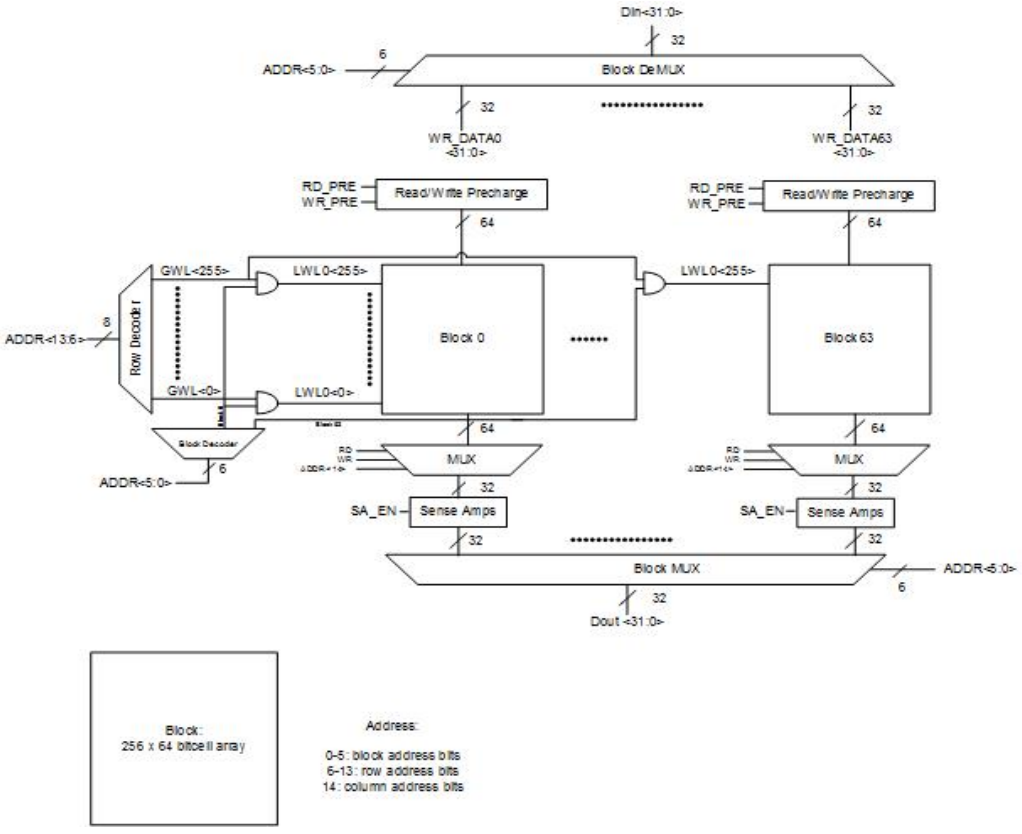


Figure 1. Global Block Diagram

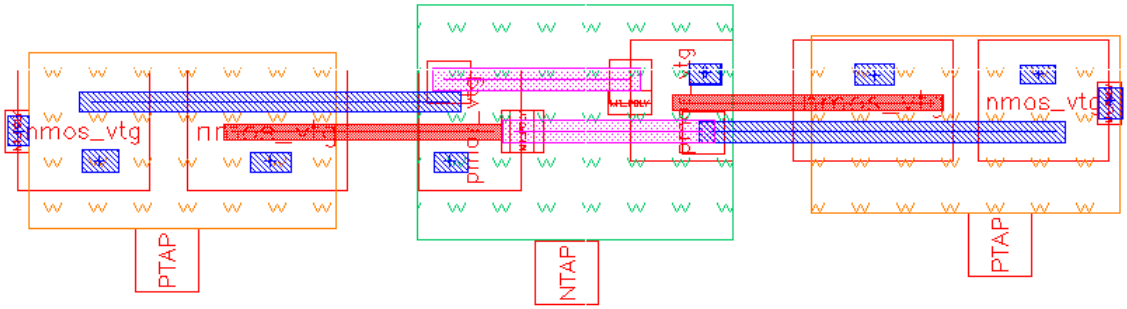


Figure 2: Proposed layout [2], passed DRC test

Transistor	VTH0 (from model specification i.e. *.inc files)
NMOS_VTL	0.3220 V
NMOS_VTG	0.4106 V
NMOS_VTH	0.6078 V

Table 1

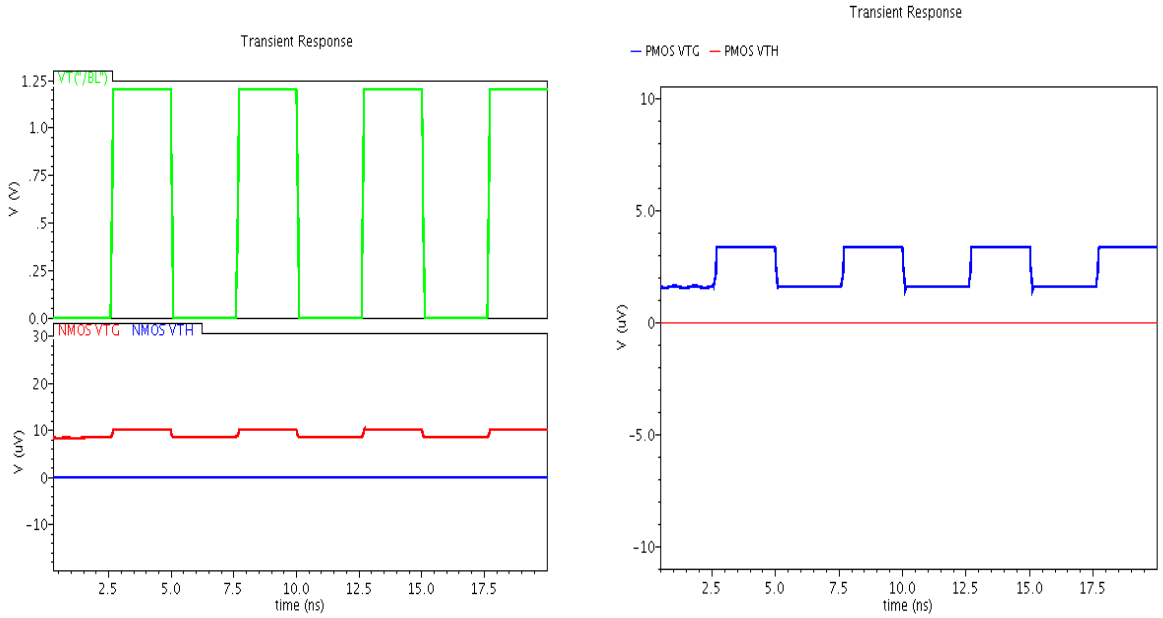


Figure 3a (top left) The voltage of the first graph shows the bitline (due to precharging). The wordline and the data line are both 0, so the bit line and the bit inverse line switches according to the precharge. However, there is leakage across the pass-transistors even during cutoff. The graphs below shows that as the threshold voltage increases (VTG to VTH), the current decreases (lower voltage in the 2nd graph across the inverting NMOS). The effect due to DIBL is shown to be insignificant (when bitline is high). Figure 3b (top right) shows similar happenings for varying the threshold voltage for PMOS.

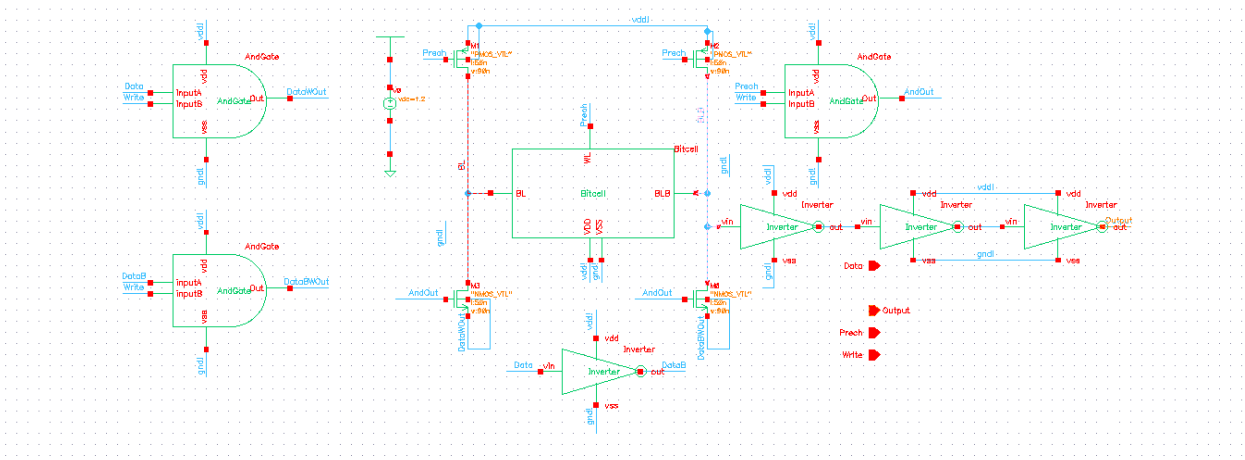


Figure 4: SRAM testbench

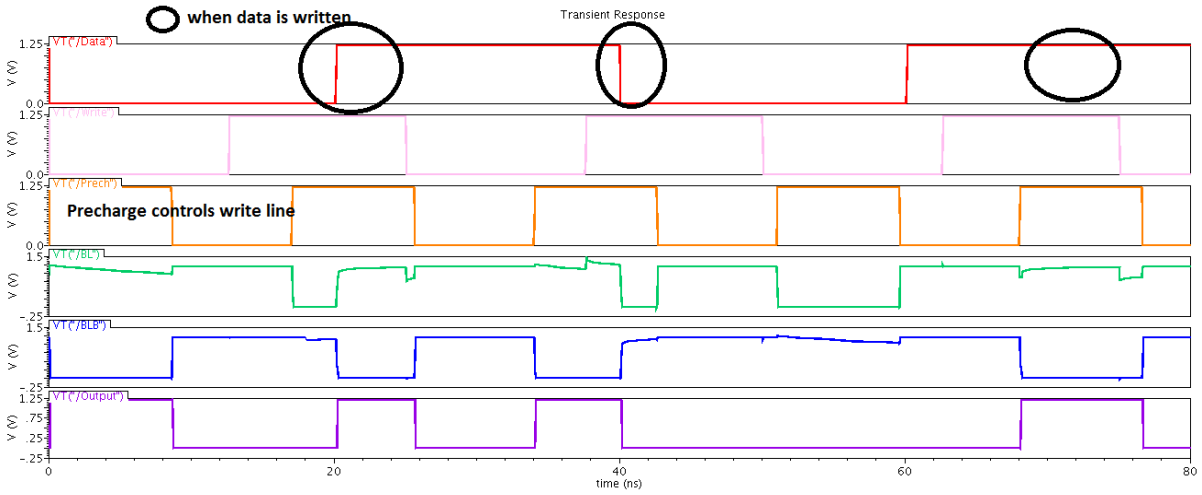


Figure 5: Working SRAM Simulation

The schematics for the simulation is in figure 4(very similar to problem set's question). Precharge controls the write line to the bitcell as there is only 1 bit (no address).Hence, when precharge is on, is write is not on, SRAM is in read mode (output reads whatever is in BL). If write is on at the same time, then it is writing whatever data is to BL (and data bar to BLB)depending on which line is 1.

Sense amplifier is not included in this simulation. The portions where BL and BLB are not exactly VDD is because of 1) parasitic capacitances (especially near the beginning when 1 is being stored, so BL slowly decreases) 2)when precharge /write happens to rise/fall (and overcharge parasitic capacitors). 3) NMOS from data is not good at writing a 1.